Réseaux — Partie 4 : Travaux pratiques

IUT de Villetaneuse — R&T 1ère année

6 mai 2012

1 Commandes de base

Objectif : L'objectif de ce TP est de se familiariser avec les commandes de base de gestion des interfaces réseau sous Linux (`ifconfig`, `ping`, `route`, `arp`) et d'étudier les formats des protocoles `IP`, `ICMP`, `ARP`.

1 La couche réseau TCP/IP

Trois principaux protocoles constituent la couche réseau TCP/IP :

`IP` (Internet Protocol) assure l'adressage, le routage de paquets et la gestion de la fragmentation ainsi que l'assemblage des paquets ;

`ICMP` (Internet Control Message Protocol) est utilisé pour véhiculer les messages de contrôle et de gestion ;

`ARP` (Address Resolution Protocol) permet de retrouver sur une liaison de donnée, telle qu'Ethernet, l'adresse physique d'une machine (l'adresse `MAC`) à partir d'une adresse logique (l'adresse `IP`).

1.1 Adresses `IPv4`

Une *adresse* `IPv4` est une suite de 32 bits (soit 4 octets) composée de deux parties :

`Net-ID` qui désigne l'adresse du réseau ;

`Host-ID` qui indique l'adresse d'une machine connectée à ce réseau.

Pour faciliter la lecture des adresses `IPv4`, celles-ci sont notées sous la forme de 4 nombres correspondant aux 4 octets et séparés par des points.

Exemple : L'adresse `00001010 00000000 00000000 00000001` est notée `10.0.0.1`.

1.2 Masques réseau

Une structure annexe appelée *masque réseau* (`netmask`) permet de distinguer les deux parties d'une adresse `IPv4`. C'est une suite de 32 bits également, dans laquelle les bits correspondant à la partie `Net-ID` sont à `1` et tous les autres (correspondant donc à `Host-ID`) sont à `0`.

Étant données une adresse a et un masque m, l'adresse réseau est obtenue en effectuant un `et` bit à bit entre a et m.

Exemple : Soient $a =$`10.0.0.1` et $m =$`255.0.0.0`. L'adresse réseau est alors `10.0.0.0`.

Une autre manière de noter l'adresse et le masque est d'utiliser `adresse IP/n` où n est le nombre de bits utilisés pour la partie `Net-ID`.

Exemple : L'adresse précédente peut être également notée `10.0.0.1/8`.

1.3 Routage

La fontion de *routage* a pour rôle de trouver un chemin (encore appelé *route*) reliant la machine source d'un paquet `IP` à la machine destination. Le réseau TCP/IP utilise la technique de *commutation de paquets* avec laquelle deux paquets envoyés d'une machine m_1 à une machine m_2 peuvent suivre des chemins différents et arriver dans un ordre différent de leur ordre d'émission.

L'approche de routage la plus simple (mais la moins performante) est le *routage statique* : chaque machine traversée par les paquets possède une *table de routage* composée d'un ensemble ordonné de *règles statiques de routage* ayant chacune la forme suivante :

`destination passerelle masque drapeaux interface`

Pour effectuer l'évaluation d'une règle de routage, on effectue un `et` de l'adresse destination d'un paquet à transmettre et du masque de la règle de routage à évaluer. Si le résultat est égal au champ destination de la règle, celle-ci est utilisée et le paquet envoyé sur l'interface de cette règle. Si le paquet doit être envoyé à un routeur, l'adresse de celui-ci figure dans le champ passerelle de la règle de routage, et le champ drapeaux comporte le drapeau `G` (`Gateway`). Sinon, il est envoyé à la machine destination et le drapeau `H` (`Host`) est positionné. Un drapeau `U` (`Up`) indique que la règle de routage est active.

Si la valeur obtenue par l'opération `et` n'est pas celle du champ destination de la règle de routage, celle-ci ne peut pas être appliquée. Toutes les règles sont examinées une par une jusqu'à temps d'en trouver une valide. Si aucune règle ne peut être appliquée, un paquet `ICMP` est envoyé à la machine source pour lui notifier l'incident.

Exemple : Soit la table de routage suivante, d'une machine connectée à un réseau local dont l'adresse est `192.168.1.0/24` :

Destination	Passerelle	Masque	Drapeaux	Interface
192.168.1.0	-	255.255.255.0	U	eth1
default	192.168.1.254	0.0.0.0	UG	eth1

La première règle indique qu'un paquet à destination d'une machine connectée sur le même réseau local est envoyé directement à la machine destination. La seconde règle indique que tout autre paquet sera envoyé à un routeur d'adresse `192.168.1.254`. L'ordre des règles dans la table de routage est important : en effet, la première règle valide est utilisée.

2 Commandes de base

2.1 ifconfig

La commande `ifconfig` permet de configurer et consulter la configuration des interfaces réseau d'une machine. Le tableau suivant en indique quelques exemples d'utilisation.

Commande	Description
ifconfig -a	Afficher l'état de toutes les interfaces
ifconfig eth0	Afficher la configuration de l'interface eth0
ifconfig eth0 10.0.0.1/8	Attribuer l'adresse 10.0.0.1/8 à l'interface eth0
ifconfig eth0 up	Activer l'interface eth0
ifconfig eth0 down	Désactiver l'interface eth0
ifconfig eth0 mtu 1000	Fixer la taille maximale des trames à 1000 octets

De nombreuses autres options sont disponibles, que l'on peut obtenir en consultant le manuel en ligne.

2.2 ping

La commande `ping` permet d'envoyer des paquets `ICMP` de type `ECHO-Request` à une machine identifiée par son adresse `IP`. À la réception d'un tel paquet, la machine destination répond par un paquet `ICMP` de type `ECHO-Reply`. La commande `ping` permet donc de tester la connectivité d'une machine à partir d'une autre, tout en donnant des informations sur le trafic entre les deux machines en mesurant le temps d'aller/retour. Elle dispose également de nombreuses options telles que préciser le nombre de paquets `ICMP` à envoyer.

Exemple : La commande :

```
ping 10.0.0.1 -c 1 -s 500 -D
```

envoie un unique paquet `ICMP` de taille 500 octets à la machine d'adresse `10.0.0.1` en positionnant le bit `DF`.

2.3 `route`

La commande `route` permet d'afficher et d'éditer la table de routage d'une machine. Des exemples typiques d'utilisation sont les suivants :

`route` affiche la table de routage.

`route add -net 10.10.0.0 netmask 255.255.0.0 eth0` ajoute la règle suivante à la fin de la table de routage :

Destination	Passerelle	Masque	Drapeaux	Interface
10.10.0.0	-	255.255.0.0	U	eth0

`route add default gw 10.10.0.1 eth0` ajoute la règle suivante à la fin de la table de routage :

Destination	Passerelle	Masque	Drapeaux	Interface
default	10.10.0.1	0.0.0.0	UG	eth0

2.4 `arp`

La commande `arp` permet de consulter et déditer la table de correspondances apprises entre les adresses logiques (`IP`) et les adresses physiques (`MAC`).

Exercice 1.1 : Câblage de l'architecture physique

Réaliser deux réseaux locaux LAN_1 d'adresse `192.168.N.0/24` et LAN_2 d'adresse `10.N.0.0/16`, où `N` est le numéro de votre rangée. Dans ce réseau, 4 machines sont utilisées :

PC_1 est connectée au réseau LAN_1 ;

PC_2 est connectée aux deux réseaux et joue le rôle de routeur ;

PC_3 **et** PC_4 sont connectées au réseau LAN_2.

Exercice 1.2 : Configuration du réseau

Question 1 : Proposer un plan d'adressage et attribuer les adresses proposées aux différentes machines. Par convention, on attribue à un routeur connecté à un réseau la dernière adresse disponible sur ce réseau. Tester le bon fonctionnement de chaque réseau séparément l'un de l'autre.

Question 2 : Afficher les tables de routage de chacune des machines et justifier leur contenu.

Question 3 : Utiliser la commande `hostname` pour attribuer à chaque machine son adresse symbolique PC_i. Le fichier `/etc/hosts` permet de spécifier des correspondances entre adresses symboliques et adresses `IP`. Éditer ce fichier sur chacune des machines de sorte à pouvoir utiliser les adresses symboliques des autres machines.

Question 4 : À l'aide de `wireshark`, capturer le trafic échangé sur le réseau LAN_1 suite à l'exécution de la commande `ping pc2 -c 1` sur la machine PC_1. Donner les éléments suivants et les justifier :

- le chronogramme des trames échangées entre PC_1 et PC_2 ;

– le format d'un paquet ARP et le fonctionnement de ce protoole ;
– le format d'un paquet ICMP ;
– le contenu de la table arp de la machine PC_1.

Question 5 : Capturer le trafic échangé sur le réseau LAN_1 suite à l'exécution de la commande `ping pc3 -c 1` sur la machine PC_1.

Question 6 : Par défaut, une machine LINUX ne route pas les paquets IP reçus. Afin d'activer le routage sur une machine, un drapeau `ip_forward` doit être positionné à la valeur 1. Ce drapeau se trouve dans le fichier `/proc/sys/net/ipv4/ip_forward`. Écrire un script shell nommé `routeur` qui fonctionne comme suit :

`routeur` affiche l'état courant du drapeau `ip_forward` ;

`routeur on` active le routage ;

`routeur off` désactive le routage ;

`routeur -h` affiche un message d'aide expliquant l'utilisation de cette commande.

Question 7 : Utiliser le script `routeur` pour activer le routage sur PC_2. Exécuter la commande `ping pc3 -c 1` sur la machine PC_1. Justifier le résultat obtenu.

Question 8 : En utilisant la commande `route`, éditer les tables de routage des différentes machines pour établir des chemins entre les deux réseaux. Tester le bon fonctionnement de votre configuration.

Question 9 : Exécuter la commande `ping pc3 -c 1` sur la machine PC_1 et capturer les trames qui circulent sur les deux réseaux suite à cette exécution. Donner le chronogramme d'échange de trames et de paquets sur les trois machines PC_1, PC_2 et PC_3. Relever les adresses source et destination. Justifier le trafic ainsi observé.

Question 10 : Insérer l'adresse MAC du routeur dans la table arp des machines PC_1, PC_3 et PC_4. Exécuter la commande ping précédente. Quels sont les effets des modifications des tables arp ?

Question 11 : Le paquet IP envoyé de PC_1 à PC_3 a-t-il subi des modifications en traversant le routeur PC_2 ? Justifier votre réponse.

2 Prise en main de Marionnet

Objectif : Marionnet est un laboratoire de manipulation de réseaux virtuels. Les objectifs du TP sont d'une part de prendre en main cet outil, et d'autre part de refaire le point sur l'interconnexion de réseaux locaux et le routage statique.

Introduction à Marionnet

Marionnet est un logiciel libre de manipulation de laboratoire virtuel de réseaux (voir le site `www.marionnet.org`). Ce logiciel permet d ?exécuter sur une seule machine des machines Linux virtuelles représentant des machines que l ?on peut interconnecter pour réaliser des réseaux locaux interconnectés en utilisant les équipements virtuels suivants : des câbles droits ou croisés, des hubs, des commutateurs et des routeurs. L ?interaction avec une machine virtuelle Linux se passe classiquement via un terminal exécutant un interpréteur shell.

Petit manuel d'utilisation

Marionnet offre une interface graphique assez intuitive qui permet de réaliser des montages de réseaux. Pour com- mencer un montage ouvrez un nouveau projet. Des icônes qui représentent les principaux équipements utilisables seront alors disponibles. En cliquant sur une icône vous avez la possibilité d'ajouter un équipement désigné au montage, d'activer, de désactiver ou d'éditer les caractéristiques d'un équipement existant. L'onglet `interfaces` permet de configurer les interfaces réseaux des équipements.

Une fois le montage est terminé, vous pouvez l'activer en cliquant sur le bouton `tout démarrer`. Des fenêtres représentant les différents équipements seront alors affichées.

L'accès à une machine se fait via un terminal exécutant un interpréteur shell qui porte le nom de la machine. Par défaut on utilise le login/mot de passe : `root`/`root`.

Vous avez la possibilité d'arrêter et de démarrer un équipement via des menus contextuels qui apparaissent en cliquant sur l'icône qui représente l'équipement cible.

Exercice 2.1 :

On souhaite réaliser le montage de la figure 1. Ce montage comporte 3 rangées, chacune avec 3 machines PC_{i1}, PC_{i2} et PC_{i3}, où i est le numéro de la rangée. De plus :
- les machines PC_{i1}, PC_{i2} sont connectées à un réseau LAN_{i1} d'adresse `192.169.i.0/24` ;
- les machines PC_{i2}, PC_{i3} sont connectées à un réseau LAN_{i2} d'adresse `10.i.0.0/16` ;
- un réseau réseau commuté transversal relie les trois machines $PC_{i1}, \forall i \in [1, 3]$.

Question 1 : Ouvrir un nouveau projet Marionnet et réaliser le montage de la figure 1 en faisant attention au choix des types de câbles à utiliser pour connecter les différents équipements. Attribuer des adresses `IP` valides aux différentes interfaces et configurer les machines PC_{i2} pour jouer le rôle d'un routeur entre les deux réseaux d'une rangée. Tester votre montage et justifier la validité de votre configuration.

Question 2 : Configurer les tables de routage des machines PC_{i1} de sorte à assurer la connectivité entre les machines PC_{i2}. Tester votre configuration.

Question 3 : Sur la machine PC_{22}, exécuter la commande `ping pc23 -n 1`. Quel est le résultat de cette commande ? Observer et expliquer le trafic généré par l'exécution de cette commande.

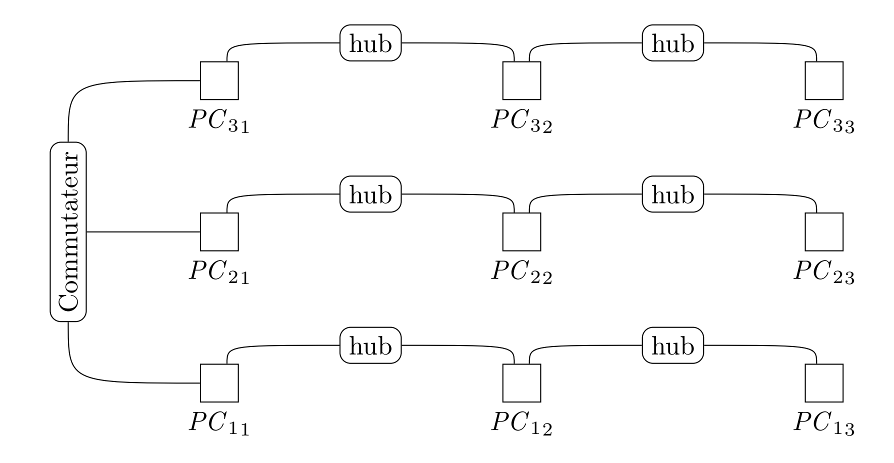


Figure 1 – Schéma de câblage

Question 4 : Sur la machine PC_{11}, exécuter la commande `ping pc33 -n 1`. Quel est le résultat de cette commande ? Observer et expliquer le trafic généré par l'exécution de cette commande.

3 Routage hiérarchique

Exercice 3.1 :

On considère un réseau d'adresse `130.120.0.0/16`. Ce réseau est relié à Internet par un routeur R. On souhaite le décomposer en deux sous-réseaux, chacun d'eux étant décomposé en trois sous-réseaux. Chacun de ces derniers sous-réseaux doit permettre d'accueillir au plus 24 machines.

Question 1 : Proposer un plan d'adressage pour les différents sous-réseaux, en précisant la première et la dernière adresse disponible ainsi que le masque associé.

Question 2 : Utiliser Marionnet pour réaliser ce montage en se limitant à représenter chaque sous-réseau par les routeurs qui le relient aux autres réseaux. Configurer les tables de routages des différents routeurs pour assurer la connectivité entre les différents réseaux, tout en minimisant le nombre de règles de routage dans les différentes tables. Proposer un plan de test pour votre configuration.

Question 3 : Proposer des scénarios pour générer des paquets `ICMP` qui signalent différents types d'erreurs de routage.

4 Routage statique avec zebra

Objectifs : Ce TP a pour but de présenter zebra, un routeur logiciel multi-protocoles qui offre une interface utilisateur similaire aux routeurs Cisco.

Introduction à zebra

Le logiciel libre zebra (voir http://www.zebra.org) est un routeur logiciel multi-protocoles modulaire. Il permet d'exécuter la plupart des protocoles de routage IP sur une machine Linux, la transformant ainsi en routeur. Il comporte un démon par protocole supporté (par exemple RIP, OSPF) et un démon central qui assure le routage statique et gère également les tables de routage mises à jour par les protocoles de routage dynamique. Dans l'environnement Marionnet, les routeurs sont mis en œuvre en utilisant des machines Linux exécutant des démons zebra. Le démon zebra est associé par défaut au port TCP 2601. Ainsi, après le démarrage d'un routeur, on peut se connecter à l'interface d'administration en utilisant une connexion telnet à partir d'une machine connectée directement au routeur. Il ne faut pas oublier d'attribuer, avant le démarrage, une adresse IP valable au port du routeur connecté à la machine à partir de laquelle celui-ci est administré.

Connexion à un routeur

Soit un routeur R connecté à une machine M. Pour administrer le routeur R à partir de la machine M, on exécute la commande :

```
telnet @R 2601
```

où @R est l'adresse IP du routeur. Un mot de passe est requis pour cette connexion (avec Marionnet, le mot de passe est zebra). Si l'authentification réussit, on entre dans le mode de consultation symbolisé par un message d'invite de la forme Routeur> où Routeur est le nom donné au routeur.

Modes d'exécution

Il existe deux principaux modes de fonctionnement :

- le *mode de consultation*, qui est le mode par défaut à la connexion, permet de consulter l'état du routeur ;
- le *mode privilégié* permet de modifier la configuration et le fonctionnement du routeur. Un autre mot de passe est requis (dans le cas de Marionnet, c'est le même). Ce mode est caractérisé par un message d'invite de la forme Routeur#.

Les commandes générales sont les suivantes :

enable	passage du mode consultation au mode privilégié
disable	passage du mode privilégié au mode consultation
?	aide en ligne concernant le stade courant
list	liste des commandes disponibles au stade courant
exit ou quit	terminer le mode courant et retourner au mode précédent

Commandes de consultation

La principale commande de consultation est la commande show, disponible à la fois en mode de consultation et en mode privilégié. Elle a de nombreuses options dont la liste est obtenue par show ? et les principales sont :

`show interface`	affiche les informations sur les interfaces réseau du routeur
`show ip route`	affiche la table de routage
`show ip forwarding`	affiche l'état du drapeau de routage du noyau système
`show history`	affiche l'historique de la session

Dans le mode privilégié, la commande `write terminal` permet d'afficher la configuration courante du routeur.

Commandes de configuration

Afin de pouvoir changer la configuration du routeur, il faut d'abord entrer dans le mode privilégié en utilisant la commande `enable`, puis passer en mode terminal de configuration à l'aide de la commande `configure terminal`. Dans ce mode, le message d'invite à la forme `Routeur(config)#`.

La configuration des interfaces du routeur nécessite de passer dans un mode de configuration d'interfaces, accessible à partir du mode terminal de configuration par la commande :

```
Routeur(config)# interface nom_interface
```

Ce mode de configuration d'interfaces est caractérisé par le message d'invite `Routeur(configif)#`.

L'ajout d'une règle de routage s'effectue avec :

```
ip route réseau_destination interface_du_routeur
```

Après avoir modifié la configuration, il faut la sauvegarder :

```
Routeur# copy runningconfig startupconfig
```

Exemple : Attribuons à l'interface `eth0` l'adresse `10.0.0.1/8`.

```
Routeur# configure terminal
Routeur(config)# interface eth0
Routeur(configif)# ip addr 10.0.0.1/8
Routeur(configif)# no shut
Routeur(configif)# end
Routeur(config)# end
Routeur# copy runningconfig startupconfig
Routeur#
```

Exercice 4.1 :

Question 1 : Utiliser l'environnement Marionnet pour réaliser un montage reliant une machine PC_1 à un routeur R_1. Le routeur R_1 a au moins deux ports. La machine PC_1 a pour adresse `192.168.1.1/24`. Quelle doit être l'adresse du port de R_1 utilisé pour connecter PC_1 ? Quel type de câble utiliser pour cette connexion ? Tester la connectivité entre PC_1 et R_1 à l'aide de la commande `ping`.

Question 2 : Utiliser la machine PC_1 pour administrer le routeur R_1. Se connecter à R_1 par `telnet` et tester les commandes de consultation.

Question 3 : Relier une nouvelle machine PC_2 d'adresse `194.194.1.1/24` à R_1. Modifier les configurations des machines et du routeur pour assurer la connectivité entre les deux réseaux.

Question 4 : Ajouter un nouveau routeur R_2 connecté à R_1 et à une nouvelle machine PC_3 d'adresse `80.85.0.1/16`. Faire le nécessaire pour assurer la connectivité entre les trois machines.

Question 5 : Proposer des scénarios utilisant la commande `ping` pour capturer des paquets `icmp` signalant les problèmes de routage suivants :
- hôte inconnu (`host unreachable`) ;
- réseau inconnu (`network unreachable`) ;
- expiration de la durée de vie ;
- fragmentation requise mais non autorisée.

5 Configuration dynamique avec DHCP

Objectif : Ce TP est une initiation à la configuration dynamique de la couche IP en utilisant le protocole DHCP.

DHCP

DHCP est un protocole opérant au niveau application dont l'objectif est de fournir un service de configuration automatique de la couche IP des machines connectées sur un réseau local. Le service fonctionne en mode cient/serveur. Le *serveur* DHCP est exécuté sur une machine (généralement un routeur). Il attend des requêtes de configuration envoyées par des *clients* DHCP sur toutes ses interfaces réseau (par défaut).

Serveur DHCP

Les informations nécessaires pour attribuer et gérer les contrats de configuration des machines clientes sont sauvegardées par le serveur dans deux principaux fichiers :

`dhcpd.conf` est le fichier principal de configuration du serveur. On y précise par exemple :
- `subnet` : sous-réseaux à configurer ;
- `range` : plage d'adresses à attribuer à un sous-réseau ;
- `default-lease-time` et `max-lease-time` : durée de vie d'un contrat de configuration ;
- `routers` : adresses des routeurs par défaut ;
- serveurs de noms, noms de domaines, etc.

Sur une machine de l'environnement MARIONNET, ce fichier se trouve dans `/etc/dhcp3/dhcpd.conf`. Il contient des exemples de configurations type, commentés par des `#`.

`/var/lib/dhcp/dhcpd.leases` contient des informations sur les contrats DHCP en cours. Un serveur ne peut s'exécuter sans ce fichier. Il peut être créé avant le premier lancement du fichier en utilisant la commande :

```
touch /var/lib/dhcp/dhcpd.leases
```

Le fichier `/var/lib/dhcp/dhcpd.leases~` contient les contrats DHCP expirés.

Après l'édition du fichier `dhcpd.conf`, un redémarrage du serveur DHCP doit être effectué afin de prendre en compte la nouvelle configuration. Les commandes de démarrage et d'arrêt du serveur sur une machine MARIONNET sont les suivantes :

`/etc/init.d/dhcp3-server start`	démarrer le serveur
`/etc/init.d/dhcp3-server stop`	arrêter le serveur
`/etc/init.d/dhcp3-server restart`	redémarrer le serveur

Client DHCP

Pour demander une configuration au serveur, une machine exécute un client DHCP avec la commande `dhclient`. Le client utilise également un fichier de configuration, qui, dans le cas de MARIONNET, se trouve dans `/etc/dhcp3/dhclient.conf`. Celui-ci ne doit pas être modifié. Les contrats de configuration obtenus par le client sont sauvegardés dans `/var/lib/dhcp3/dhclient.leases`.

Exercice 5.1 :

En utilisant l'environnement MARIONNET, définir 5 machines PC_1 à PC_5. La machine PC_3 joue le rôle d'un routeur qui relie un réseau LAN_1, composé de PC_1 et PC_2 et un réseau LAN_2, composé de PC_4 et PC_5. Réaliser le montage et la configuration.

Question 1 : Installer un serveur `DHCP` sur la machine PC_3 afin de configurer les machines des deux réseaux locaux comme suit : LAN_1 est un réseau privé, d'adresse `10.0.0.0/8` ; LAN_2 a pour adresse `198.160.45.0/24`.

Question 2 : Vérifier qu'aucun serveur `DHCP` n'est lancé ailleurs que sur PC_3.

Question 3 : Activer le serveur `DHCP`. Démarrer un client sur la machine PC_1. Vérifier le bon fonctionnement du service `DHCP` et analyser les trames échangées entre PC_1 et PC_3 lors de la configuration automatique.

Question 4 : Quelle est la durée du bail concernant l'adresse de PC_1 ?

Question 5 : Exécuter des clients `DHCP` sur les autres machines et vérifier le bon déroulement des configurations.

Question 6 : Modifier la configuration du serveur afin d'attribuer toujours la même adresse à PC_1.

Question 7 : Proposer un scénario pour générer un paquet d'acquittement négatif de la part du serveur `DHCP`.